



UNITED STATES PATENT AND TRADEMARK OFFICE

MN
UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/829,370	04/22/2004	Houman Pournasseh	003797.00920	7092
28319 7590 06/04/2007 BANNER & WITCOFF, LTD. ATTORNEYS FOR CLIENT NOS. 003797 & 013797 1100 13th STREET, N.W. SUITE 1200 WASHINGTON, DC 20005-4051			EXAMINER KHATRI, ANIL	
			ART UNIT 2191	PAPER NUMBER
			MAIL DATE 06/04/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/829,370	Applicant(s) POURNASSEH ET AL.	
	Examiner Anil Khatri	Art Unit 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 22 April 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-24 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-24 is/are rejected. =
- 7) ☐ Claim(s) _____ is/are objected to. =
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement. =

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 22 April 2004 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date <u>7/9/04, 1/31/06</u> . | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Specification

Applicant is reminded of the proper language and format for an abstract of the disclosure.

The abstract should be in narrative form and generally limited to a single paragraph on a separate sheet within the range of 50 to 150 words. It is important that the abstract not exceed 150 words in length since the space provided for the abstract on the computer tape used by the printer is limited. The form and legal phraseology often used in patent claims, such as "means" and "said," should be avoided. The abstract should describe the disclosure sufficiently to assist readers in deciding whether there is a need for consulting the full patent text for details.

The language should be clear and concise and should not repeat information given in the title. It should avoid using phrases which can be implied, such as, "The disclosure concerns," "The disclosure defined by this invention," "The disclosure describes," etc.

The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

The following title is suggested: "*Language Localization and Intercepting Data Using Translation Tables*".

Claim Objections

Claims are objected to because of the following informalities: Abbreviation "GDI" should be spelled out at least once. Appropriate correction is required.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-22 are rejected under 35 USC 101 because they disclose a claimed invention that is an abstract idea as defined in the case *In re Warmerdam*, 33, F 3d 1354, 31 USPQ 2d 1754 (Fed. Cir. 1994).

Analysis: Claims 1-22 disclosed by the applicant as being a "process of modifying information...". Since the claims are each a series of steps to be performed on a computer the processes must be analyzed to determine whether they are statutory under 35 USC 101.

Examiner interprets that claims 1-22 are non-statutory because claim recites computer program product are program, per se i.e. the description or expressions of the program are not physical things nor are they statutory process as they do not act being performed. Computer programs do not define any structural and functional interrelationship between the computer program and other claimed aspect of the invention which permits the computer program's functionality could be realized. Therefore, computer program is merely a set of instructions

Art Unit: 2191

capable of being executed by a computer, the computer program itself is not a process. Thus claims 1-22 are non-statutory and rejected under 35 USC 101.

Examiner interprets that the claims 1-22 are non-statutory because claim is a computer program for processing set of instructions which capable of being intercepting, comparing and replacing executed by a computer implemented method, the computer program itself is not a process and without the computer-readable storage medium so its functionality can be realized. Applicant submit no substance that how this will be processed without incorporating a processor, memory and medium. Therefore, claims 1-22 are merely a manipulating of data, comparing and replacing data without produce a useful results and practical application. Thus claims 1-22 are non-statutory and rejected under 35 USC 101.

Analysis: Claims 23-24 disclosed by the applicant as being a “system comprising a processor...”. Since the claims are each a series of steps to be performed on a computer the processes must be analyzed to determine whether they are statutory under 35 USC 101.

Examiner interprets that claims 23-24 are not limited to tangible embodiments. In view of applicant’s disclosure, specification page 5, paragraph 23, the medium is not limited to tangible embodiments, instead being defined as including both tangible embodiments (e.g., [computer readable medium]) and intangible embodiments (e.g., [transmission media, radio frequency (RF), infrared (IR), a carrier wave, telephone line, a signal, etc.]). As such, the claim is not limited to statutory subject matter and is therefore non-statutory. To overcome this type of 101 rejection the claims need to be amended to include only the physical computer media and not a transmission media or other intangible or non-functional media. For the specification at the bottom, carrier medium and transmission media would be not statutory but storage media would be statutory.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-24 are rejected under 35 U.S.C. 102(b) as being anticipated by *Buzbee* USPN 5,909,578.

Regarding claims 1, 12 and 23

Buzbee teaches

intercepting data destined for one of a system resource or GDI (figure 3, column 7, lines 16-17, translating a code block from the native application beginning at the program counter address to produce a translated code block);

-comparing intercepted data against data in a core translation table (column 4, lines 50-60, AMT 218 is a table containing a mapping between addresses in the native application and corresponding blocks of translated code in the translated application. If a block has been translated, then AMT 218 will contain a pointer to a location within block information table 220. Block information table 220, in turn, contains a pointer the translated code block within

Art Unit: 2191

translated code cache 208. At this point in the execution, however, the table lookup in AMT 218 will fail since the application has not yet been translated. This failure is shown by the "no" branch from box 214 to box 216);

replacing intercepted data with data from core translation table based on comparing step (column 5, lines 15-25, While translating the code block, DTS 200 also checks to see whether the targets of a terminating branch statement can be replaced with the addresses of translated code blocks. Typically, the execution of a basic code block will terminate at a branch instruction. Either the branch will be taken or the code will fall through to the next block. Therefore, it is helpful to think of the terminating branch as having a taken target and a not taken target. When DTS 200 translates the code block, DTS 200 checks whether the targets of the terminating branch instruction have already been translated. If so, then DTS 200 inserts the address of the translated block into the instruction. Otherwise, DTS 200 modifies the branch to return to box 212).

Regarding claims 2 and 13

Buzbee teaches

redirecting intercepted data to one of system resource or GDI based on comparing step (figure 2, column 5, lines 35-46, next, the translated code block 222 is stored in translated code cache 208. Note that boxes 228 and 230 are discussed below with respect to FIG. 3. Then, DTS 200 executes code block 222. Since code block 222 is the first translated block, neither of its terminating branch targets have been translated. Therefore, DTS 200 goes back to box 212 when it reaches the terminating branch of code block 222).

Art Unit: 2191

Regarding claims 3 and 14

Buzbee teaches

comparing intercepted data against data in an application translation table (column 4, lines 50-60, AMT 218 is a table containing a mapping between addresses in the native application and corresponding blocks of translated code in the translated application. If a block has been translated, then AMT 218 will contain a pointer to a location within block information table 220. Block information table 220, in turn, contains a pointer the translated code block within translated code cache 208. At this point in the execution, however, the table lookup in AMT 218 will fail since the application has not yet been translated. This failure is shown by the "no" branch from box 214 to box 216); and

replacing said intercepted data with data from application translation table (column 5, lines 15-25, While translating the code block, DTS 200 also checks to see whether the targets of a terminating branch statement can be replaced with the addresses of translated code blocks. Typically, the execution of a basic code block will terminate at a branch instruction. Either the branch will be taken or the code will fall through to the next block. Therefore, it is helpful to think of the terminating branch as having a taken target and a not taken target. When DTS 200 translates the code block, DTS 200 checks whether the targets of the terminating branch instruction have already been translated. If so, then DTS 200 inserts the address of the translated block into the instruction. Otherwise, DTS 200 modifies the branch to return to box 212).

Art Unit: 2191

Regarding claims 4 and 15

Buzbee teaches

Simplifying and normalizing intercepted data (column 5, lines 1-13, In box 216, DTS 200 translates the native code block into a new block of code that performs the same function as the native block. In addition, DTS 200 instruments the new code block with profiling instructions. By using this type of dynamic instrumentation, a wide variety of profile information can be gathered. For example, different instrumentation code can be included to perform: opcode counting, branch prediction modeling, trace selection, and cache modeling. Further, the type of information gathered can be changed dynamically based upon previously gathered information. For example, the profiler could start off with simple profiling to see what code is being executed most frequently. Then, the translations could be discarded and new translations build which gather more detailed information).

Regarding claims 5 and 16

Buzbee teaches

Unifying a case of intercepted data (column 5, lines 60-67, Consider, for example, code block 222 in FIG. 2. Since code block 222 has been translated and instrumented with profiling code, the instructions within code block 222 do not necessarily correspond to the instructions within the native code block. Therefore, if the timer signal arrives while DTS 200 is in the middle of executing code block 222, the state of DTS 200 is unknown because it does not precisely map to a state in the native application).

Art Unit: 2191

Regarding claims 6 and 17

Buzbee teaches

Removing control character (column 1, lines 53-64, The above and other needs are met by a method and system for burst profiling an application. Burst profiling is the technique of gathering profile information in "bursts." In general, the profiled application runs free and unfettered, but is periodically stopped and then resumed with inserted instrumentation code. After a short period of time, the application is stopped again, the instrumentation code is removed, and unfettered execution resumes until it is time for the next burst. This technique is lightweight in that most of the time the application runs free and at full speed. Yet this technique yields more kinds of and more detailed information than sampling because it takes instrumentation traces rather than single-point snapshots).

Regarding claims 7 and 18

Buzbee teaches

Cross referencing intercepted data between resources loader and GDI (column 4, lines 33-42, FIG. 2 is a functional block diagram of the DTS 200. Note that certain boxes in FIG. 2 represent steps performed by DTS 200 while other blocks represent logical components within DTS 200. Specifically, boxes 208, 210, 212, 214, 228, and 230 represent steps performed by DTS 200. In contrast, blocks 218, 220, 222, 224, and 226 represent logical components of DTS 200. DTS 200, itself, can be integrated into the OS or a separate application on the computer system. Regardless of how it is implemented, DTS 200 functions like a dynamic loader in that it executes when required).

Art Unit: 2191

Regarding claims 8 and 19

Buzbee teaches

restoring translated data into a format of intercepted data (column 5, lines 28-34, Next, the translated code block 222 is stored in translated code cache 208. Note that boxes 228 and 230 are discussed below with respect to FIG. 3. Then, DTS 200 executes code block 222. Since code block 222 is the first translated block, neither of its terminating branch targets have been translated. Therefore, DTS 200 goes back to box 212 when it reaches the terminating branch of code block 222).

Regarding claims 9 and 20

Buzbee teaches

Resizing a displayed item to show translated data (column 4, lines 61-67, Accordingly, DTS 200 moves to box 216. At box 216, DTS 200 retrieves a block of code from the native application beginning at the instruction identified by the PC. A preferred embodiment of the present invention retrieves a basic block of application code. However, any granularity of block size can be used. In general, the block size should be selected such that it can be efficiently translated by DTS 200.

Regarding claims 10 and 21

Buzbee teaches

comparing intercepted data against data in a community-built translation table (column 4, lines 50-60, AMT 218 is a table containing a mapping between addresses in the native application and corresponding blocks of translated code in the translated application. If a block has been

Art Unit: 2191

translated, then AMT 218 will contain a pointer to a location within block information table 220. Block information table 220, in turn, contains a pointer the translated code block within translated code cache 208. At this point in the execution, however, the table lookup in AMT 218 will fail since the application has not yet been translated. This failure is shown by the "no" branch from box 214 to box 216); and

replacing intercepted data with data from community-built translation table(column 5, lines 15-25, While translating the code block, DTS 200 also checks to see whether the targets of a terminating branch statement can be replaced with the addresses of translated code blocks. Typically, the execution of a basic code block will terminate at a branch instruction. Either the branch will be taken or the code will fall through to the next block. Therefore, it is helpful to think of the terminating branch as having a taken target and a not taken target. When DTS 200 translates the code block, DTS 200 checks whether the targets of the terminating branch instruction have already been translated. If so, then DTS 200 inserts the address of the translated block into the instruction. Otherwise, DTS 200 modifies the branch to return to box 212).

Regarding claims 11 and 22

Buzbee teaches

Processing intercepted data using machine translation (column 3, lines 66-67 and column 4, lines 1-4, At step 107, the signal handler arms the timer to again fire after the predetermined interval. Then, at step 108, the signal handler passes the snapshot of the machine state to the DTS. The DTS is discussed in more detail below. At this point, however, the dynamic translator can be considered a generic machine simulator that does profiling).

Art Unit: 2191

Regarding claim 24

Buzbee teaches

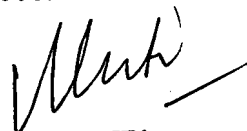
A storage that stores at least one core translation table storage being accessed by processor to obtain translated data (figures 2-4, column 4, lines 27-32).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anil Khatri whose telephone number is 571-272-3725. The examiner can normally be reached on M-F 8:30-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Zhen can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.


ANIL KHATRI
PRIMARY EXAMINER